TU

UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/650,238	08/27/2003	Wolfram Schulte	3382-65592	6392

26119   7590   01/19/2007

KLARQUIST SPARKMAN LLP
121 S.W. SALMON STREET
SUITE 1600
PORTLAND, OR 97204

EXAMINER
VU, TUAN A

ART UNIT	PAPER NUMBER
2193	

SHORTENED STATUTORY PERIOD OF RESPONSE	MAIL DATE	DELIVERY MODE
3 MONTHS	01/19/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

PTOL-90A (Rev. 10/06)

	Application No.	Applicant(s)
Office Action Summary	10/650,238	SCHULTE ET AL.
	Examiner	**Art Unit**	
	Tuan A. Vu	2193	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE _3_ MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1) ☒ Responsive to communication(s) filed on _8/27/03_.

2a) ☐ This action is **FINAL**.   2b) ☒ This action is non-final.

3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4) ☒ Claim(s) _1-33_ is/are pending in the application.

    4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5) ☐ Claim(s) _____ is/are allowed.

6) ☒ Claim(s) _1-33_ is/are rejected.

7) ☐ Claim(s) _____ is/are objected to.

8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9) ☐ The specification is objected to by the Examiner.

10) ☒ The drawing(s) filed on _27 August 2003_ is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.

    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

    Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

    a) ☐ All   b) ☐ Some * c) ☐ None of:

      1. ☐ Certified copies of the priority documents have been received.

      2. ☐ Certified copies of the priority documents have been received in Application No. _____.

      3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

    * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☒ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _2/20/04; 6/08/05_.

4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____.

5) ☐ Notice of Informal Patent Application

6) ☐ Other: _____.

DETAILED ACTION

1.   This action is responsive to the application filed 8/27/2003.

Claims 1-33 have been submitted for examination.

Claim Rejections - 35 USC § 101

2.   35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

3.   Claims 1-33 are rejected under 35 U.S.C. 101 because the claimed invention is directed

to non-statutory subject matter.

Specifically, claims 1, 19, and 33 recite computer method comprising receiving domain

configuration information, a reflection of a program, and producing data domain based on the

configuration information and reflection. What is conveyed from the producing step is that some

domain is generated from a computer method, with no reasonable teaching that this domain data

amounts to any real world usefulness of any form. As a whole, the claimed subject matter

amounts to an abstract and necessarily computer-based internal data (being the end result)

because one skill in the art applying the computer method thus claimed cannot reasonably

perceive that a real world application result has been generated from acknowledging such

domain data being produced, particularly in terms of useful, concrete and tangible computer

output.

> The Federal Circuit has recently applied the practical application test in determining whether the claimed subject matter is statutory under 35 U.S.C. § 101. The practical application test requires that a " useful, concrete, and tangible result" be accomplished. An "abstract idea" when practically applied is eligible for a patent. As a consequence, an invention, which is eligible for patenting under 35 U.S.C. § 101, is in the "useful arts" when it is a machine, manufacture, process or composition of matter, which produces a concrete, tangible, and useful result. The test for practical application is thus to determine whether the claimed invention produces a "useful, concrete and tangible result".

The claimed *data domain* being produced therefore does not reasonably convey the realization of a practical application result, which is required to be concrete, and necessarily useful and tangible. Further, claims 19 and 33 recite a system without providing a tangible hardware support to realize the functionality of the configuration manager being recited, hence are further deficient in terms of statutory subject matter.

In all, claims 1, 19 and 33 are rejected for leading to a non-statutory subject matter.

Claims 2-18 for not specifying further on the useful aspect of the data domain will be rejected for leading to a non-statutory subject matter.

Claims 20-32 are also rejected for the same reasons as above.

Claim Rejections - 35 USC § 112

4.   The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

5.   Claim 18 is rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

Claim 18 recites the limitation "data domains of the parameter", such that the parameter occurs for the first time with respect to the base claims. There is insufficient antecedent basis for this limitation in the claim; and will be treated as a data structure element or method.

Claim Rejections - 35 USC § 102

6.   The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

7.   Claims 1-33 are rejected under 35 U.S.C. 102(b) as being anticipated by Davidson et al.,

USPN: 6,083,276(hereinafter Davidson).

        As per claim 1, Davidson discloses a computer implemented method for producing a

data domain for a data structure element of a computer program, the method comprising:

        receiving domain configuration information corresponding to the data structure element

(e.g. Fig. 3A, 3B, 3C);

        receiving a reflection of the computer program (e.g. bean objects 212- Fig. 2); and

        producing the data domain (e.g. Fig. 4B, 4C – Note: mapping corresponding descriptor or

attribute for a method or class reads on data domain combining configuration information with

reflection of beans components) based on the domain configuration information and the program

reflection.

        As per claim 2, see Fig. 1 for computer readable media having computer executable

instructions for performing the method of claim 1.

        As per claim 3, refer to claim 1 for a listing of data structure elements of the computer

program as reflection of computer program (e.g. Fig. 2).

        As per claim 4, Davidson discloses annotating code of the computer program (e.g.

comment 302 – Figs. 3; e.g. col. 29-36 Appendix A for ADML for comments between special

tags <!-- ... -->) with the domain configuration information.

        As per claims 5-6, Davidson discloses computer readable media having computer

executable instructions for compiling the code of the computer program annotated with the

domain configuration information for producing the data domain (Fig. 4B, 4C cols 21-28) according to its domain configuration information.

As per claims 7-8, Davidson discloses the domain configuration information comprising one or more expressions (e.g. BML – col. 8-col. 10 – Note: tag specification *<Foo Att1 = Value 1 Att2=Value2 ... />* reads on explicit denotation of domain to be produced) for explicitly denoting the data domain to be produced corresponding in form to one that is applicable to the data structure element; wherein the expressions comprise methods and functions (e.g. parameter method Fig. 4C; METHOD, ARGUMENTS - col. 16, lines 28-42; Fig. 5; *CALL calls a method ... Attributes* - Appendix A, col. 47, bottom - Note: beans constructs being described in BML language as method and arguments reads on methods and functions) defined within the code of the computer program, which are exposed via the reflection of the computer program.

As per claim 9, Davidson discloses wherein the data structure element is a data type with one or more fields and the form of the explicitly expressed data domain is a set of values of the fields comprising the data type (e.g. TYPE – col. 16, lines 54-63).

As per claim 10, Davidson discloses wherein the data structure element is a method (re claim 9) and the form of the explicitly expressed data domain is a set of tuples of parameters (e.g. *GET FIND CONSTANT|ARRAY| ANY| ALL|NOT –* Appendix A, col. 29-30) of the method.

As per claim 11, Davidson discloses wherein the data structure element is a field or a parameter of a designated type (Table 2, pg. 19; TYPE ID - lines 55-63, col. 16) and the form of the explicitly expressed data domain is an enumeration of values (e.g. <VALUE ...</VALUE> lines 55-63, col. 16; lines 28-42, col. 16) of the designated type corresponding to the field or the parameter.

As per claim 12, Davidson discloses inheriting (e.g. Fig. 3C; *children* - col. 10, lines 50-65; *Child component ...Parent component* – Fig. 4D; lines 9-29 - col. 13) the data domain to be produced from the data domain of other related data structure elements.

As per claim 13, Davidson discloses a data type comprising a plurality of sub-types and a selection of one or more of the plurality of sub-types wherein the data domain to be produced for the data type is a union of data domains of the sub-types (P tag ... Table 2, col. 19; Style tag...Table 3, col. 20 – Note: paragraph and style tag with subtypes read on plurality of subtype and selection from an union of subtypes) belonging to the selection.

As per claim 14, Davidson discloses data structure element being a field or a parameter of a designated type (Fig. 5; Table 2, pg. 19; TYPE ID - lines 55-63, col. 16; cols. 17-18) and the domain configuration information comprises information indicating that the data domain to be produced for the field or the parameter is inherited (e.g. Fig. 3C; *children* - col. 10, lines 50-65; *Child component... Parent component* – Fig. 4D; lines. 9-29 - col. 13)from the data domain of their designated type.

As per claims 15-16, Davidson discloses domain configuration information related to producing the data domain for the data structure element by applying domain generation techniques on other selected data domains; and filtering the result of the applying domain generation technique step using a predicate (steps 410, 418, Fig. 4B; Match 424, Conversion 432, More attributes 440 – Fig. 4C; step 456, Fig. 4D – Note: code to map components as called by description information with respect to parse algorithm reads on predicates for filtering data from information domain into data domain – see Java pseudo-code col. 26, 28).

As per claims 17-18, Davidson discloses data structure element is a data type with a

plurality of fields (lines 30-34, 55-58 -col. 9; lines 30-43, col. 16; Table 2, col. 19) and the other

data domains are data domains of the fields (re claim 1 or Fig. 4B, 4C); wherein the data

structure element is a method (Table 1, col. 17-18) and the other data domains are data domains

of the parameter (re claim 1).

As per claim 19, Davidson discloses a system for producing a data domain for a data

structure element of a computer program, the system comprising:

a domain configuration manager for receiving domain configuration information (e.g.

3A, 3B, 3C) corresponding to the data structure element and

using a reflection of the computer program (e.g. bean objects 212- Fig. 2) to produce the

data domain for the data structure element according to the domain configuration information

(Fig. 4B, 4C – Note: mapping corresponding descriptor or attribute for a method or class reads

on data domain combining configuration information with reflection of beans components).

As per claims 20-21, Davidson discloses a graphical user interface communicative with

the domain configuration manager for receiving the domain configuration information and

transferring (Fig. 1; col. 18-40 -col.28; Error Message – Fig 4B) the domain configuration

information to the domain configuration manager; a GUI for receiving user input related to the

domain configuration information (e.g. user and application-generated 'events' lines 15-40 -col.

10).

As per claim 22, Davidson discloses domain configuration manager for reading the

reflection of the computer program to identify the data structure element for its domain

configuration (e.g. Fig. 3B, 3C; Fig. 4B).

As per claim 23, Davidson discloses wherein the data structure element is a data type and the domain configuration manager is operable for producing the data domain for the data type according to an explicit expression indicative of the data domain of the data type (refer to rationale of claims 13-14).

As per claim 24, Davidson discloses wherein the explicit expression comprises methods and functions defined within the computer program (e.g. col. 17-18) and exposed to the domain configuration manager via the reflection of the computer program (Table 1, col. 17-18).

As per claim 25 Davidson discloses wherein the data structure element is a method and the domain configuration manager is operable for producing the data domain as a set of tuples of parameters of the method according to an explicit expression of the domain configuration information (refer to claim 10).

As per claim 26 Davidson discloses wherein the data structure element is a field or a parameter of a declared type and the data configuration manager is operable for producing the data domain according to an explicit expression whose result is an enumeration of values of the declared type (refer to claim 11).

As per claim 27 Davidson discloses wherein the data structure element is a data type with sub-types and the data configuration manager is operable for producing the data domain for the data type through inheritance as a union of data domains of its selected sub-types (refer to claim 13).

As per claim 28 Davidson discloses wherein the data structure element is a data type and the data configuration manager is operable for producing the data domain for the data type by

applying a domain generation technique to one or more fields of the data type (refer to claim
15).

As per claim 29, Davidson discloses wherein the domain generation technique is a
Cartesian product (e.g. *mapper 122, 124* – Fig. 1; *map 418* – Fig. 4b; *Mach 424*, Fig. 4C;
BeanInfo Mapper - Fig. 5 - Note: a Cartesian or Cross product between 2 sets A and B is defined
as the set of all pairs {*a, b*} such that *a* is an element of the set A and *b* is an element of the set B;
i.e. mapping an element of A with a corresponding element of B) of the selected fields of the
data type and the domain configuration manager is further operable for applying a constraint
specified in the domain configuration information to the Cartesian product for producing the data
domain for the data type (refer to claim 16 for filtering constraint using predicate).

As per claim 30 Davidson discloses wherein the data structure element is a field or a
parameter of a declared type and the domain configuration manager is operable for producing
the data domain for the field or the parameter as the data domain of their respective declared type
through inheritance (refer to claim 14).

As per claim 31, Davidson discloses wherein the data structure element is a method (re
claim 24-25) and the domain configuration manager is operable for producing the data domain
for the method by applying a domain generation technique to the parameters of the method (re
claim 15).

As per claim 32, Davidson discloses wherein the domain configuration technique is a
Cartesian product (re claim 29) of the data domains of the parameters (re claims 26 and 30) of
the method and the data configuration manager is further operable for applying a constraint (refer

to claim 16) for filtering constraint using predicate) to the result of the Cartesian product for

producing the data domain for the data type.

As per claim 33, Davidson discloses a system for producing data domains of data

structure elements of a computer program, the system comprising means for:

receiving domain configuration information corresponding to the data structure

elements;

reading a reflection of the computer program; and

processing the domain configuration information and the reflection to produce the data

domains corresponding to the data structure elements;

all of which limitations having been addressed in claim 1.

Conclusion

8.   Any inquiry concerning this communication or earlier communications from the

examiner should be directed to Tuan A Vu whose telephone number is (272) 272-3735. The

examiner can normally be reached on 8AM-4:30PM/Mon-Fri.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's

supervisor, Meng-Ai An can be reached on (571)272-3756.

The fax phone number for the organization where this application or proceeding is

assigned is (571) 273-3735 (for non-official correspondence - please consult Examiner

before using) or 571-273-8300 (for official correspondence) or redirected to customer

service at 571-272-3609.

Any inquiry of a general nature or relating to the status of this application should be

directed to the TC 2100 Group receptionist: 571-272-2100.

Information regarding the status of an application may be obtained from the Patent

Application Information Retrieval (PAIR) system. Status information for published applications

may be obtained from either Private PAIR or Public PAIR. Status information for unpublished

applications is available through Private PAIR only. For more information about the PAIR

system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR

system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Tuan A Vu
Patent Examiner,
Art Unit 2193
January 15, 2007